



Lee Sullivan Shea & Smith LLP

Cole B. Richter | Attorney

656 W Randolph St
Floor 5W
Chicago, Illinois 60661

312.757.4478 Office
richter@ls3ip.com

January 5, 2022

VIA CM/ECF

The Honorable William Alsup
United States District Court Judge
United States District Court for the Northern District of California
450 Golden Gate Avenue
San Francisco, CA 94102

Re: Google LLC v. Sonos, Inc., Case No. 3:20-cv-6754-WHA (N.D.CA)

Dear Judge Alsup:

Google has asked this Court to: (1) compel Sonos to identify the functionality and source code it is mapping to the “local playback queue” element; (2) limit Sonos to the specific “media-item identifiers” that it identified in its contentions; and (3) to strike “certain products” in Sonos’s contentions not identified by name and model number.

Regarding (1), Sonos disagrees that it has failed to identify the functionality and source code it is mapping to the “local playback queue”. In particular Sonos has identified what Google refers to as a “Receiver SDK” as implementing the local playback queue. Google’s documentation explains that a “Receiver SDK” (which is implemented on an accused media player) “maintains a queue” which may include one or more items.¹ Because these items are maintained “to play sequentially on the receiver,” they form a “local playback queue.”

Accordingly, there is no dispute here that Google’s media players include a “local playback queue” or that Sonos has identified *what* it contends implements that queue.

Google argues that Sonos has not identified (with enough specificity) which portions of the media player source code implement that queue. But, Sonos has identified the source code that runs on the accused media players that, based on its investigation to date, creates the receiver side queue. For the **YouTube, YouTube Music, YouTube TV, and YouTube Kids apps**, Sonos identified, *inter alia*, [REDACTED]; and [REDACTED] (Ex. A at 58-59). For the **Play Music app**, Sonos identified, *inter alia*, [REDACTED]; and [REDACTED]

¹ See, e.g., https://developers.google.com/cast/docs/android_sender/queueing (“*The Receiver SDK maintains the queue and responds to operations on the queue as long as the queue has at least one item currently active (playing or paused).*”) (Ex. A at 49).



Lee Sullivan Shea & Smith LLP

Cole B. Richter | Attorney

656 W Randolph St
Floor 5W
Chicago, Illinois 60661

312.757.4478 Office
richter@ls3ip.com

[REDACTED] (Ex. A at 63). For the **Podcasts app**, Sonos identified, *inter alia*, [REDACTED]; [REDACTED]; and [REDACTED] (Ex. A at 67). For the **Cast-Enabled Displays**, Sonos identified, *inter alia*, [REDACTED]; and [REDACTED] (Ex. A at 72).

This is source code that runs on the accused media players and that Sonos believes is responsible for [REDACTED] – the local queuing functionality Google’s own documents acknowledge. *See* Ex. A at 50-51. Once the media player receives the data [REDACTED] (as a result of this code being executed), Sonos believes the media player [REDACTED] – thus creating the queue. *See id.*

Although Sonos has spent several months diligently analyzing Google’s source code, the code is both complex and gargantuan and Sonos has not yet identified all of the portions that are involved in storing the media items once they are pulled down to the player. Indeed, there are separate codebases for each accused app and the accused media players, each with several hundreds of thousands of lines of code, as well as several hundreds of thousands of lines of relevant remote server and web-services code. To provide more details about the “local playback queue” limitation in the code, Sonos needs more discovery from Google. To date, however, Google has not cooperated with Sonos’s discovery attempts. For example, on August 7, 2021², Sonos served an interrogatory asking for Google to explain how its products play back a sequence of media items. On September 7, rather than offer any explanation, Google cited under Rule 33(d) to its entire source code production (which is many hundreds of thousands of lines) and the entirety of a 48,000+ page document production (effectively saying “the answer is in there somewhere”). Sonos raised this issue with Google on September 23. On December 23 (after it filed its letter brief), Google supplemented its response to this interrogatory, not by providing a narrative response, or even citing to certain source code files, but by reducing the 48,000+ page citation to a citation of 20,000+ pages.

Although written discovery has not been helpful on the matter, Sonos fully expects that a deposition of Google’s software engineers on the relevant source code will provide Sonos with the discovery it needs to supplement its infringement contentions with respect to the “local playback queue” limitation. Sonos has served Google with a Rule 30(b) deposition in this regard, and hopes to have this deposition completed in the next few weeks.

Notwithstanding the above, in an effort to reduce the number of disputes before

² This is the date fact discovery opened when the case was pending in the Western District of Texas prior to transfer here.



Your Honor, Sonos can agree to Google’s request for Sonos to provide a supplement within 14 days that further details the functionality and source code that Sonos contends amounts to the “local playback queue” element as Sonos presently understands that element. Given that Google is requesting this additional detail, and has not provided meaningful discovery responses in responses to Sonos’ requests, good cause exists for allowing Sonos to provide it.

Regarding (2), Sonos’s use of “e.g.” or other open-ended language to identify certain “media-item identifiers” was not an attempt to shoe-horn in additional unidentified identifiers. Rather, it was used as a recognition that discovery has just begun and Sonos has not had an opportunity to question the relevant Google engineers. Sonos can, therefore, agree to remove the “e.g.” in regard to “media-item identifiers.” But, given that Sonos still awaits an appropriate interrogatory response from Google explaining how its products work and the opportunity to question Google’s engineers, Sonos anticipates seeking leave to amend its contentions in light of that discovery.

Regarding (3), P.L.R. 3-1(b) requires an identification of “each accused apparatus, product, device, process, method, act, or other instrumentality (‘Accused Instrumentality’) of each opposing party of which the party is aware . . . Each product, device, and apparatus shall be identified by name or model number, if known.” P.L.R. 3-1(b). The Accused Instrumentalities for the patents relevant to (3) are several apps, which Sonos has identified by name. Google does not contend otherwise. For some (but not all) asserted claims, the claims require the apps to be installed onto a computer device, such as a phone or tablet. But the specific type or model of computer device on which the app is installed has, as far as we can tell, no bearing on the accused functionality because that functionality is defined by the accused apps not the make and model of the phone on which the app is installed.

Put differently, what Google is asking for here is for Sonos to give Google a list (by make and model numbers) of all of the phones on which Google’s software *can* be installed – despite the fact that (a) this information is irrelevant to understanding *what is accused* and why and (b) the fact that Google has this information (because it licenses / approves those phones) and has refused to produce it to Sonos in discovery. Despite this, Sonos did identify specific names and model numbers of Google computer devices onto which the apps may be installed. Sonos does not presently know the names or model numbers of other computer devices that end users have, themselves, actually downloaded and installed these apps onto and thus could not have identified them by name and model number. That cannot be within the spirit of P.L.R. 3-1(b) and, if required, would result in an over-inclusive list of hypothetical devices that would be unhelpful to either party in preparing its respective cases, particularly here as the discrete model of device has no bearing on the accused functionality.



Lee Sullivan Shea & Smith LLP

Cole B. Richter | Attorney

656 W Randolph St
Floor 5W
Chicago, Illinois 60661

312.757.4478 Office
richter@ls3ip.com

Respectfully submitted,

A handwritten signature in blue ink, appearing to read 'C.B.R.' followed by a stylized flourish.

Cole B. Richter
Counsel for Sonos, Inc.